IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
WACO DIVISION

ANCORA TECHNOLOGIES, INC. Plaintiff, v. VIZIO, INC., Defendant.	Civil Action No. 6:21-cv-739 Jury Trial Requested

COMPLAINT FOR PATENT INFRINGEMENT

This is an action for patent infringement in which Ancora Technologies, Inc. makes the following allegations against VIZIO, Inc. ("VIZIO"):

RELATED CASE

1. This case is related to the actions *Ancora Technologies, Inc. v. Roku, Inc.* (W.D. Tex. Jul. 16, 2021); *Ancora Technologies Inc. v. Nintendo Co. Ltd. et al.* (W.D. Tex. Jul. 16, 2021); and *Ancora Technologies Inc. v. Google, LLC, Inc.* (W.D. Tex. Jul. 16, 2021)—each of which was filed on July 16, 2021, in the United States District Court for the Western District of Texas, Waco Division, asserting infringement of United States Patent No. 6,411,941.

PARTIES

2. Plaintiff Ancora Technologies, Inc. is a corporation organized and existing under the laws of the State of Delaware with a place of business at 23977 S.E. 10th Street, Sammamish, Washington 98075.

3. Defendant VIZIO, Inc. ("VIZIO"), is a corporation organized and existing under the laws of California.

JURISDICTION AND VENUE

4. This action arises under the patent laws of the United States, Title 35 of the United States Code, such that this Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

5. This Court also has personal jurisdiction over VIZIO because directly or through intermediaries, VIZIO has committed acts within the Western District of Texas giving rise to this action and/or has established minimum contacts with the Western District of Texas such that the exercise of jurisdiction would not offend traditional notions of fair play and substantial justice.

6. For example, VIZIO has placed or contributed to placing infringing products like the VIZIO XR6M10 SmartCast Tablet into the stream of commerce via an established distribution channel knowing or understanding that such products would be sold and used in the United States, including in the Western District of Texas.

7. Further, on information and belief, VIZIO also has derived substantial revenues from infringing acts in the Western District of Texas, including from the sale and use of infringing products like the VIZIO XR6M10 SmartCast Tablet.

8. In addition, venue is proper under 28 U.S.C. § 1391(b)-(c) and 28 U.S.C. § 1400 as VIZIO also maintains authorized sellers and sales representatives through VIZIO Official Retailers such as Best Buy, Costco, Walmart, Target, and Sam’s Club (https://www.vizio.com/en/official-retailers), which offer and sell Accused Products in this District, and which maintain a regular and established place of business for VIZIO at locations such as, e.g., Best Buy at 4627 S Jack Kultgen Expy, Waco, TX 76706; Best Buy at 3550 S General Bruce Dr, Temple, TX 76504; Sam’s Club at 2301 E Waco Dr, Waco, TX 76705; Costco at 1901 Kelly Ln., Pflugerville, TX, 78660; Costco at

4601 183A Toll Rd., Cedar Park, TX 78613; Walmart at 4320 Franklin Ave, Waco, TX 76710; Walmart at 600 Hewitt Dr, TX 76712; and Walmart at 733 Sun Valley Blvd, Hewitt, TX 76643.

THE ASSERTED PATENT

9. This lawsuit asserts causes of action for infringement of United States Patent No. 6,411,941 ("the '941 Patent"), which is entitled "Method of Restricting Software Operation Within a License Limitation."

10. The U.S. Patent and Trademark Office duly and legally issued the '941 Patent on June 25, 2002.

11. Subsequent to issue, and at least by December 21, 2004, all right, title, and interest in the '941 Patent, including the sole right to sue for any infringement, were assigned to Ancora Technologies, Inc., which has held, and continues to hold, all right, title, and interest in the '941 Patent.

12. The president of Ancora Technologies, Inc.—Mr. Miki Mullor—is one of the inventors of the '941 Patent.

13. A reexamination certificate to the '941 Patent subsequently was issued on June 1, 2010.

14. Since being assigned to Ancora Technologies, Inc., the '941 Patent has been asserted in patent infringement actions filed against Microsoft Corporation, Dell Incorporated, Hewlett Packard Incorporated, Toshiba America Information Systems, Apple Inc., HTC America, Inc., HTC Corporation, Samsung Electronics Co., Ltd., Samsung Electronics America, Inc., LG Electronics, Inc., LG Electronics U.S.A., Inc., Sony Mobile Communications AB, Sony Mobile Communications, Inc., Sony Mobile Communications (USA) Inc., Lenovo Group Ltd., Lenovo

(United States) Inc., Motorola Mobility, LLC, TCT Mobile (US) Inc., and Huizhou TCL Mobile Communication Co., Ltd.

15. In the course of these litigations, a number of the ’941 Patent’s claim terms have been construed, and the validity of the ’941 Patent has been affirmed repeatedly.

16. For example, in December 2012, the United States District Court for the Northern District of California issued a claim construction order construing the terms (1) “volatile memory”; (2) “non-volatile memory”; (3) “BIOS”; (4) “program”; (5) “license record”; and (6) “verifying the program using at least the verification structure.” *Ancora Techs., Inc. v. Apple Inc.*, No. 11–CV–06357 YGR, 2012 WL 6738761, at *1 (N.D. Cal. Dec. 31, 2012).

17. Further, the court rejected Apple’s indefiniteness arguments and further held that, at least with respect to Claims 1-3 and 5-17, “[t]he steps of the Claim do not need to be performed in the order recited.” *Ancora Techs., Inc. v. Apple Inc.*, No. 11–CV–06357 YGR, 2012 WL 6738761, at *5, *13 (N.D. Cal. Dec. 31, 2012).

18. Subsequently, the United States Court of Appeals for the Federal Circuit affirmed the district court’s rejection of Apple’s indefiniteness argument. *Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732, 739 (Fed. Cir. 2014).

19. The Federal Circuit also agreed with Ancora Technologies, Inc. that “the district court erred in construing ‘program’ to mean ‘a set of instructions for software applications that can be executed by a computer’”—holding that, as Ancora had argued, the term should be accorded its normal meaning of “‘a set of instructions’ for a computer.” *Ancora Techs., Inc. v. Apple, Inc.*, 744 F.3d 732, 734-35, 737 (Fed. Cir. 2014).

20. Subsequently, in a more recent decision, the Federal Circuit held that the ’941 Patent satisfied § 101 as a matter of law—stating: “[W]e conclude that claim 1 of the ’941 patent is not

directed to an abstract idea." *Ancora Techs., Inc. v. HTC Am., Inc.*, 908 F.3d 1343 (Fed. Cir. 2018), *as amended* (Nov. 20, 2018).

21. In addition, the Patent Trial and Appeal Board rejected HTC's request to institute covered business method review proceedings on the '941 Patent—explaining that "the '941 [P]atent's solution to the addressed problem is rooted in technology, and thus, is a 'technical solution'" and also rejecting HTC's argument that "the '941 [P]atent recites a technological solution that is not novel and nonobvious."

22. This Court likewise issued a claim construction order construing or adopting the plain and ordinary meaning of various claims of the '941 Patent, including (1) "non-volatile memory"; (2) "license"; (3) "license record"; (4) "volatile memory"; (5) "BIOS"; (6) "memory of the BIOS"; (7) "program"; (8) "selecting a program residing in the volatile memory"; (9) "using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS"; (10) "set up a verification structure"; (11) "verifying the program using at least the verification structure"; (12) "acting on the program according to the verification"; (13) "first non-volatile memory area of the computer"; (14) the Claim 1 preamble; and (15) the order of Claim 1 steps. *Ancora Technologies, Inc. v. LG Electronics, Inc.*, 1:20-cv-00034-ADA, at Dkt. 69 (W.D Tex. June 2, 2020).

23. Finally, and most recently, the United States District Court for the Central District of California issued a claim construction order construing the terms (1) "volatile memory"; (2) "selecting a program residing in the volatile memory"; (3) "set up a verification structure"; (4) "license record"; (5) "memory of the BIOS"; and (6) the whole of Claim 8. *Ancora Techs., Inc v. TCT Mobile (US), Inc., et al.*, No. 8:19-cv-02192-GW-AS, ECF No. 66 & 69 (C.D. Cal. Nov. 18-19, 2020).

COUNT 1 – INFRINGEMENT

24. Plaintiff repeats and incorporates by reference each preceding paragraph as if fully set forth herein and further state:

25. VIZIO has infringed the ’941 Patent in violation of 35 U.S.C. § 271(a) by, prior to the expiration of the ’941 Patent, selling, and/or offering for sale in the United States, and/or importing into the United States, without authorization, products and/or operating system software for products that are capable of performing at least Claim 1 of the ’941 Patent literally or under the doctrine of equivalents and, without authorization, then causing such products to perform each step of at least Claim 1 of the ’941 Patent.

26. At a minimum, such Accused Products include those servers/software utilized by VIZIO to transmit an over-the-air (“OTA”) software update, as well as those televisions, streaming players, and other devices and technology that included VIZIO’s operating system software and to which VIZIO sent or had sent an OTA update that caused such device to perform the method recited in Claim 1 prior to the expiration of the ’941 Patent.

27. Such Accused Products include products like the VIZIO XR6M10 SmartCast Android Tablet Remote (“VIZIO XR6M10 SmartCast Tablet”), which—as detailed below—VIZIO configured such that it would be capable of performing each step of Claim 1 of the ’941 Patent and subsequently provided one or more OTA updates that caused the device to perform each step of Claim 1.[1]

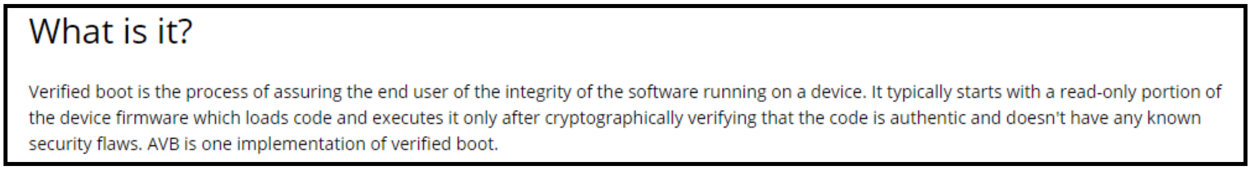
28. Such Accused Products also include products like the VIZO XR6P10, VIZIO VTAB1008-V, VIZIO Co-Star (VAP430), VIZIO ISV-B11, VIZIO ISG-B03, VIZIO M Series

[1] This description of infringement is illustrative and not intended to be an exhaustive or limiting explanation of every manner in which each Accused Product infringes the ’941 patent. Further, on information and belief, the identified functionality of the VIZIO XR6M10 SmartCast Android Tablets are representative of components and functionality present in all Accused Products.

M422i-B1, VIZIO M Series M492i-B2, VIZIO M Series M502i-B1, VIZIO M Series M552i-B2, VIZIO M Series M602i-B3, VIZIO M Series M652i-B2, VIZIO M Series M702i-B3, VIZIO M-Series M80-C3, VIZIO M-Series M75-C1, VIZIO M-Series M70-C3, VIZIO M-Series M65-C1, VIZIO M-Series M60-C3, VIZIO M-Series M55-C2, VIZIO M-Series M50-C1, VIZIO M-Series M49-C1, VIZIO M-Series M43-C1, VIZIO M-Series M50-D1, VIZIO M-Series M55-D0, VIZIO M-Series M60-D1, VIZIO M-Series M65-D0, VIZIO M-Series M70-D3, VIZIO M-Series M80-D3, VIZIO D-Series D40u-D1, VIZIO D-Series D50u-D1, VIZIO D-Series D55u-D1, VIZIO D-Series D58u-D3, VIZIO D-Series D65u-D2, VIZIO D-Series LED Smart TV D24h-E1, VIZIO D-Series D32f-E1, VIZIO D-Series D32ff1, VIZIO D-Series D39f-E1, VIZIO D-Series D40-E1, VIZIO D-Series D43f-E1, VIZIO D-Series Ultra HD D43-E2, VIZIO D-Series D48f-E0, VIZIO D-Series D50f-E1, VIZIO D-Series Ultra HD D50-E1, VIZIO D-Series D55f-E0, VIZIO D-Series Ultra HD D55-E0, VIZIO D-Series Ultra HD D65-E0, VIZIO SmartCast E-Series™ 32" , VIZIO SmartCast E-Series E43-E2, VIZIO SmartCast E-Series E50-E1, E50-E3, VIZIO SmartCast E-Series E55-E1,E55-E2, VIZIO SmartCast E-Series E60-E3, VIZIO SmartCast E-Series E65-E0, E65-E1, VIZIO SmartCast E-Series E70-E3, VIZIO SmartCast E-Series E75-E3, VIZIO SmartCast E-Series E80-E3, VIZIO M-Series M50-E1, VIZIO M-Series M55-E0, VIZIO M-Series M70-E3, VIZIO M-Series M75-E1, VIZIO P Series P55-E1, VIZIO P Series P65-E1, VIZIO P Series P75-E1, VIZIO D-Series D43-F1, VIZIO D-Series D55-F2, VIZIO D-Series D60-F3, VIZIO D-Series D65-F1, VIZIO D-Series D70-F3, VIZIO SmartCast™ E-Series™ E43-F1, VIZIO SmartCast™ E-Series™ E50-F2, VIZIO SmartCast™ E-Series™ E55-F1, VIZIO SmartCast™ E-Series™ E65-F0, VIZIO SmartCast™ E-Series™ E70-F3, VIZIO SmartCast™ E-Series™ E75-F1, VIZIO P-Series P55-F1, VIZIO P-Series P65-F1, VIZIO P-Series P75-F1, VIZIO SB3651, VIZIO SB3851, VIZIO SB2820, VIZIO SB3621, VIZIO SB3251, VIZIO SB3651, VIZIO SB3821, VIZIO SB3831, VIZIO SB3851, VIZIO SB4451, VIZIO SB4031, VIZIO SB2621, VIZIO SB3820, VIZIO SB3830, VIZIO SB4031, VIZIO SB4551, VIZIO SB3821, VIZIO SB36512, VIZIO SB46312, and VIZIO SB46514, as well as any predecessor models to such devices, to which VIZIO sent, or had sent, an OTA update prior to the expiration of the ’941 Patent.

29. For example, Claim 1 of the ’941 Patent claims “a method of restricting software operation within a license for use with a computer including an erasable, non-volatile memory area of a BIOS of the computer, and a volatile memory area; the method comprising the steps of: [1] selecting a program residing in the volatile memory, [2] using an agent to set up a verification structure in the erasable, non-volatile memory of the BIOS, the verification structure accommodating data that includes at least one license record, [3] verifying the program using at least the verification structure from the erasable non-volatile memory of the BIOS, and [4] acting on the program according to the verification.”

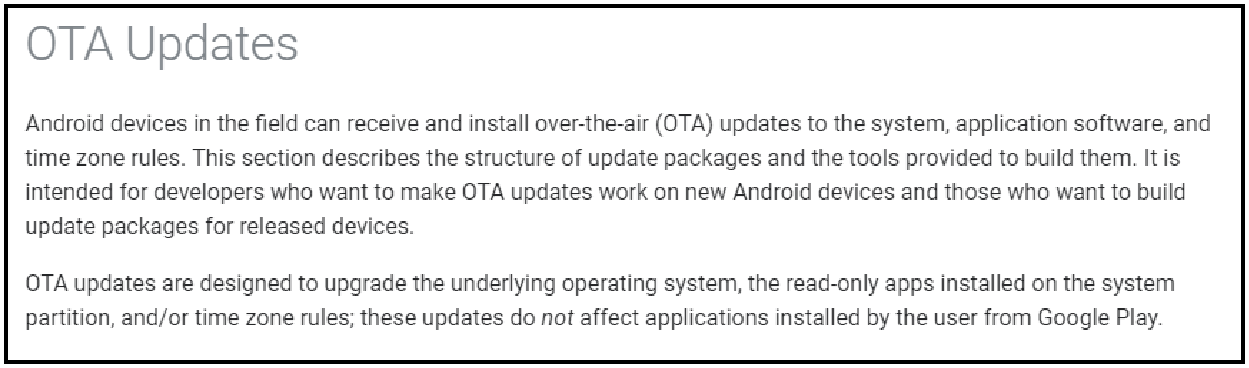
30. When VIZIO transmitted an OTA update like its XR6M10 Version 03.99.01.04, VIZIO performed and/or caused devices like the VIZIO XR6M10 SmartCast Tablet to perform each element of Claim 1 as part of its VIZIO-specified, pre-configured software update process:



What is it?

Verified boot is the process of assuring the end user of the integrity of the software running on a device. It typically starts with a read-only portion of the device firmware which loads code and executes it only after cryptographically verifying that the code is authentic and doesn't have any known security flaws. AVB is one implementation of verified boot.

https://android.googlesource.com/platform/external/avb/+/master/README.md#Build-System-Integration.

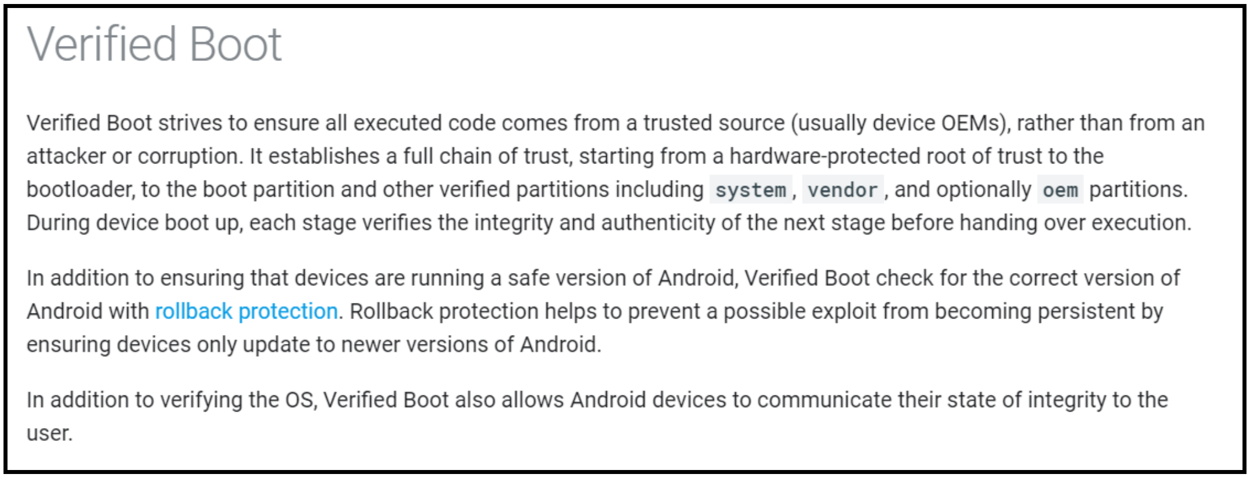


OTA Updates

Android devices in the field can receive and install over-the-air (OTA) updates to the system, application software, and time zone rules. This section describes the structure of update packages and the tools provided to build them. It is intended for developers who want to make OTA updates work on new Android devices and those who want to build update packages for released devices.

OTA updates are designed to upgrade the underlying operating system, the read-only apps installed on the system partition, and/or time zone rules; these updates do *not* affect applications installed by the user from Google Play.

https://source.android.com/devices/tech/ota.



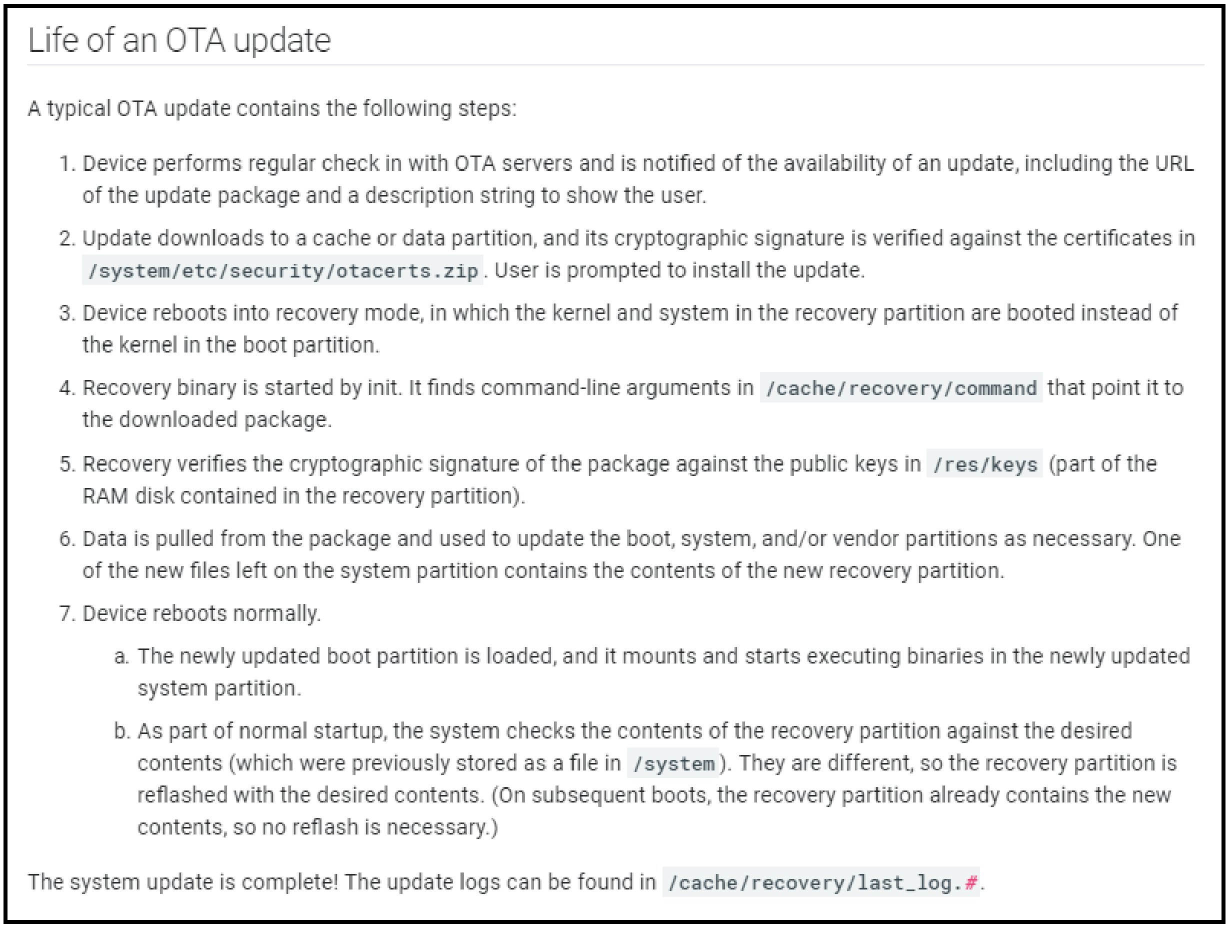
Verified Boot

Verified Boot strives to ensure all executed code comes from a trusted source (usually device OEMs), rather than from an attacker or corruption. It establishes a full chain of trust, starting from a hardware-protected root of trust to the bootloader, to the boot partition and other verified partitions including `system`, `vendor`, and optionally `oem` partitions. During device boot up, each stage verifies the integrity and authenticity of the next stage before handing over execution.

In addition to ensuring that devices are running a safe version of Android, Verified Boot check for the correct version of Android with rollback protection. Rollback protection helps to prevent a possible exploit from becoming persistent by ensuring devices only update to newer versions of Android.

In addition to verifying the OS, Verified Boot also allows Android devices to communicate their state of integrity to the user.

https://source.android.com/security/verifiedboot.



Life of an OTA update

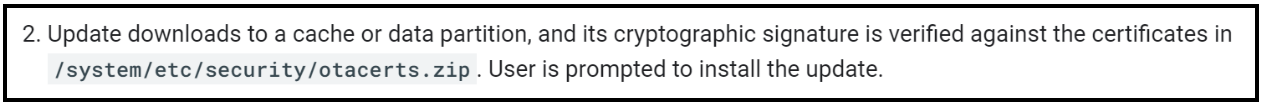
A typical OTA update contains the following steps:

1. Device performs regular check in with OTA servers and is notified of the availability of an update, including the URL of the update package and a description string to show the user.
2. Update downloads to a cache or data partition, and its cryptographic signature is verified against the certificates in `/system/etc/security/otacerts.zip`. User is prompted to install the update.
3. Device reboots into recovery mode, in which the kernel and system in the recovery partition are booted instead of the kernel in the boot partition.
4. Recovery binary is started by init. It finds command-line arguments in `/cache/recovery/command` that point it to the downloaded package.
5. Recovery verifies the cryptographic signature of the package against the public keys in `/res/keys` (part of the RAM disk contained in the recovery partition).
6. Data is pulled from the package and used to update the boot, system, and/or vendor partitions as necessary. One of the new files left on the system partition contains the contents of the new recovery partition.
7. Device reboots normally.
    a. The newly updated boot partition is loaded, and it mounts and starts executing binaries in the newly updated system partition.
    b. As part of normal startup, the system checks the contents of the recovery partition against the desired contents (which were previously stored as a file in `/system`). They are different, so the recovery partition is reflashed with the desired contents. (On subsequent boots, the recovery partition already contains the new contents, so no reflash is necessary.)

The system update is complete! The update logs can be found in `/cache/recovery/last_log.#`.

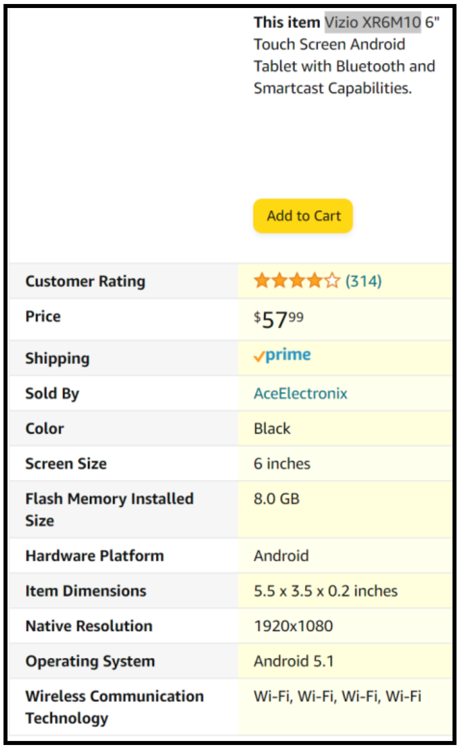
https://source.android.com/devices/tech/ota/nonab.

31. In particular, each VIZIO XR6M10 SmartCast Tablet contains both erasable, non-volatile memory in the form of flash memory and volatile memory in the form of RAM memory. Such non-volatile memory includes a cache or data partition which—on information and belief—is an example of BIOS memory:



2. Update downloads to a cache or data partition, and its cryptographic signature is verified against the certificates in `/system/etc/security/otacerts.zip`. User is prompted to install the update.

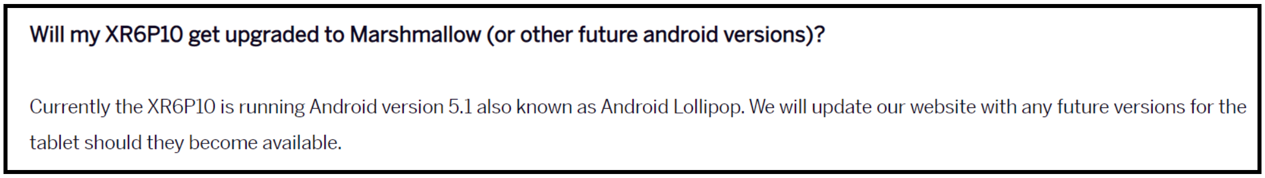
https://source.android.com/devices/tech/ota/nonab;



	This item Vizio XR6M10 6" Touch Screen Android Tablet with Bluetooth and Smartcast Capabilities. Add to Cart
Customer Rating	★★★★☆ (314)
Price	$\$57^{99}$
Shipping	✓prime
Sold By	AceElectronix
Color	Black
Screen Size	6 inches
Flash Memory Installed Size	8.0 GB
Hardware Platform	Android
Item Dimensions	5.5 x 3.5 x 0.2 inches
Native Resolution	1920x1080
Operating System	Android 5.1
Wireless Communication Technology	Wi-Fi, Wi-Fi, Wi-Fi, Wi-Fi

https://www.amazon.com/Vizio-Remote-XR6M10-SmartCast-Tablet/dp/B07BMFHLDY.

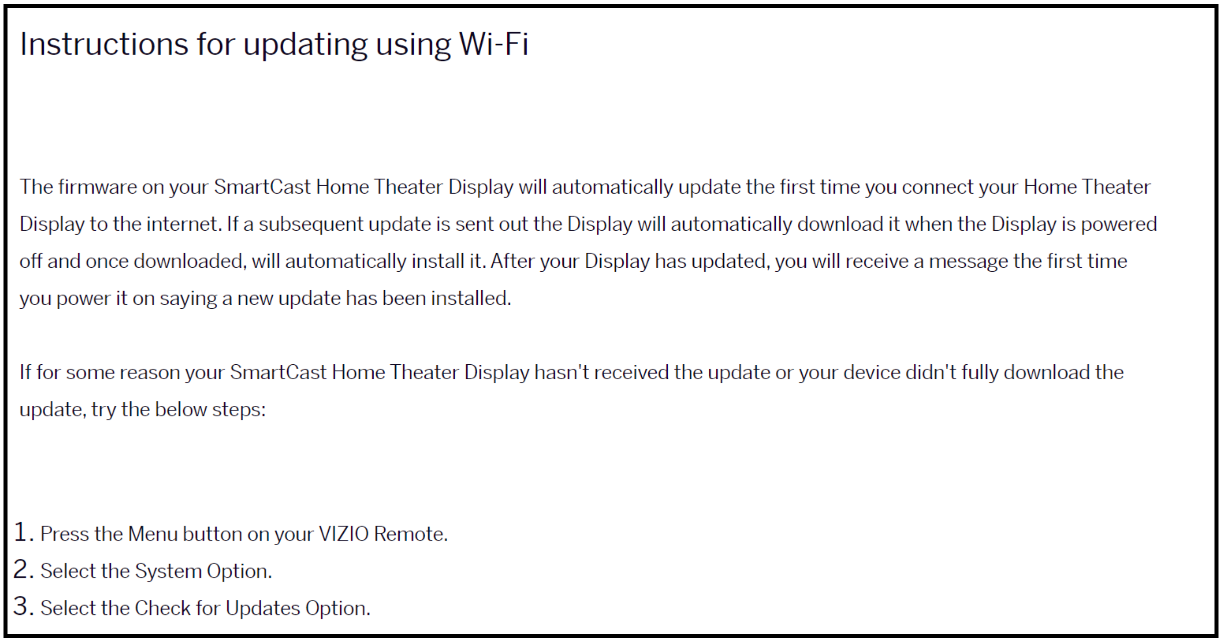
32. Further, as detailed above, each VIZIO XR6M10 SmartCast Tablet was configured by VIZIO to repeatedly check to see if a new software update was available, including through the following method:



Will my XR6P10 get upgraded to Marshmallow (or other future android versions)?

Currently the XR6P10 is running Android version 5.1 also known as Android Lollipop. We will update our website with any future versions for the tablet should they become available.

https://support.vizio.com/s/article/Firmware-Information-215?language=en_US.



Instructions for updating using Wi-Fi

The firmware on your SmartCast Home Theater Display will automatically update the first time you connect your Home Theater Display to the internet. If a subsequent update is sent out the Display will automatically download it when the Display is powered off and once downloaded, will automatically install it. After your Display has updated, you will receive a message the first time you power it on saying a new update has been installed.

If for some reason your SmartCast Home Theater Display hasn't received the update or your device didn't fully download the update, try the below steps:

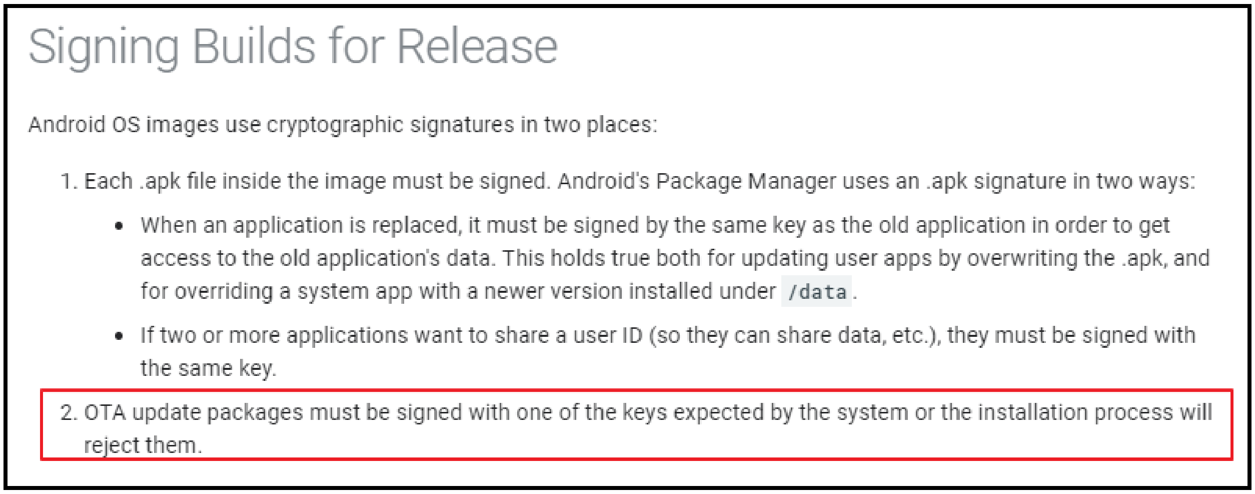
1. Press the Menu button on your VIZIO Remote.
2. Select the System Option.
3. Select the Check for Updates Option.

https://support.vizio.com/s/article/Firmware-Information-155?language=en_US.

33. During this process, one or more OTA servers owned or controlled by VIZIO set up a verification structure in the erasable, non-volatile memory of the BIOS of the VIZIO XR6M10 SmartCast Tablet by transmitting to the device an OTA update, which the VIZIO XR6M10 SmartCast Tablet is configured by VIZIO to save to the erasable, non-volatile memory of its BIOS. As noted previously, on information and belief, such BIOS areas include what VIZIO refers to as the cache or data memory area partition.

34. This OTA update contains a verification structure that, on information and belief, includes data accommodating at least one license record.

35. Examples of such a license record include what is known as a Private Key and/or a Public Key, which may be encrypted with an RSA signature:

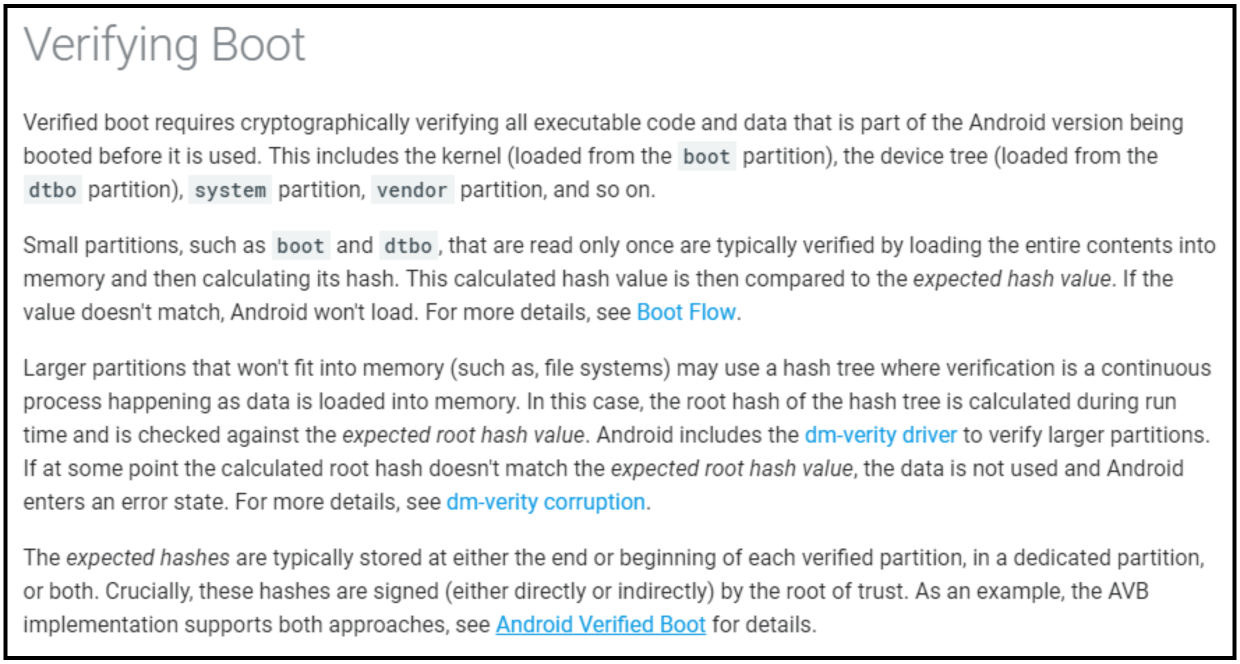


Signing Builds for Release

Android OS images use cryptographic signatures in two places:

1. Each .apk file inside the image must be signed. Android's Package Manager uses an .apk signature in two ways:
   - When an application is replaced, it must be signed by the same key as the old application in order to get access to the old application's data. This holds true both for updating user apps by overwriting the .apk, and for overriding a system app with a newer version installed under `/data`.
   - If two or more applications want to share a user ID (so they can share data, etc.), they must be signed with the same key.
2. OTA update packages must be signed with one of the keys expected by the system or the installation process will reject them.

36. https://source.android.com/devices/tech/ota/sign_builds. Other examples include a cryptographic hash or hash tree:



Verifying Boot

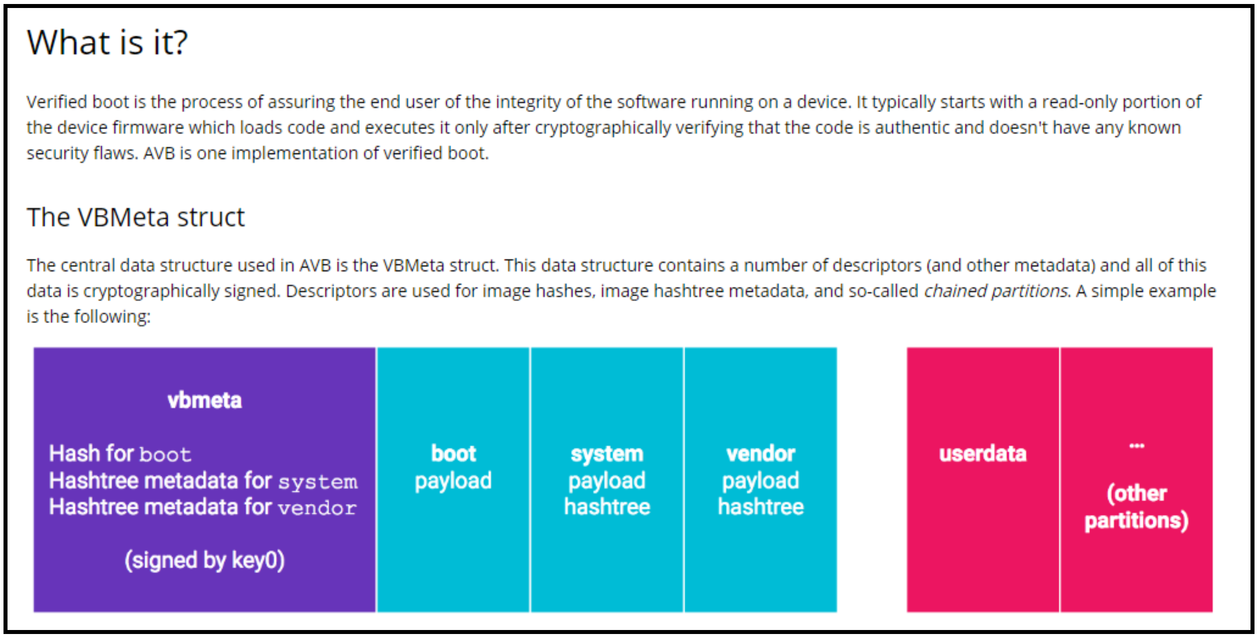
Verified boot requires cryptographically verifying all executable code and data that is part of the Android version being booted before it is used. This includes the kernel (loaded from the `boot` partition), the device tree (loaded from the `dtbo` partition), `system` partition, `vendor` partition, and so on.

Small partitions, such as `boot` and `dtbo`, that are read only once are typically verified by loading the entire contents into memory and then calculating its hash. This calculated hash value is then compared to the *expected hash value*. If the value doesn't match, Android won't load. For more details, see Boot Flow.

Larger partitions that won't fit into memory (such as, file systems) may use a hash tree where verification is a continuous process happening as data is loaded into memory. In this case, the root hash of the hash tree is calculated during run time and is checked against the *expected root hash value*. Android includes the dm-verity driver to verify larger partitions. If at some point the calculated root hash doesn't match the *expected root hash value*, the data is not used and Android enters an error state. For more details, see dm-verity corruption.

The *expected hashes* are typically stored at either the end or beginning of each verified partition, in a dedicated partition, or both. Crucially, these hashes are signed (either directly or indirectly) by the root of trust. As an example, the AVB implementation supports both approaches, see Android Verified Boot for details.

https://source.android.com/security/verifiedboot/verified-boot.

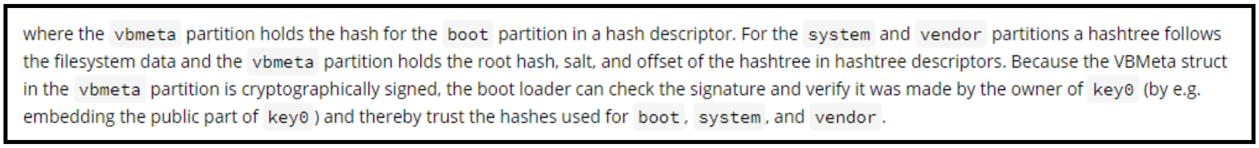


What is it?

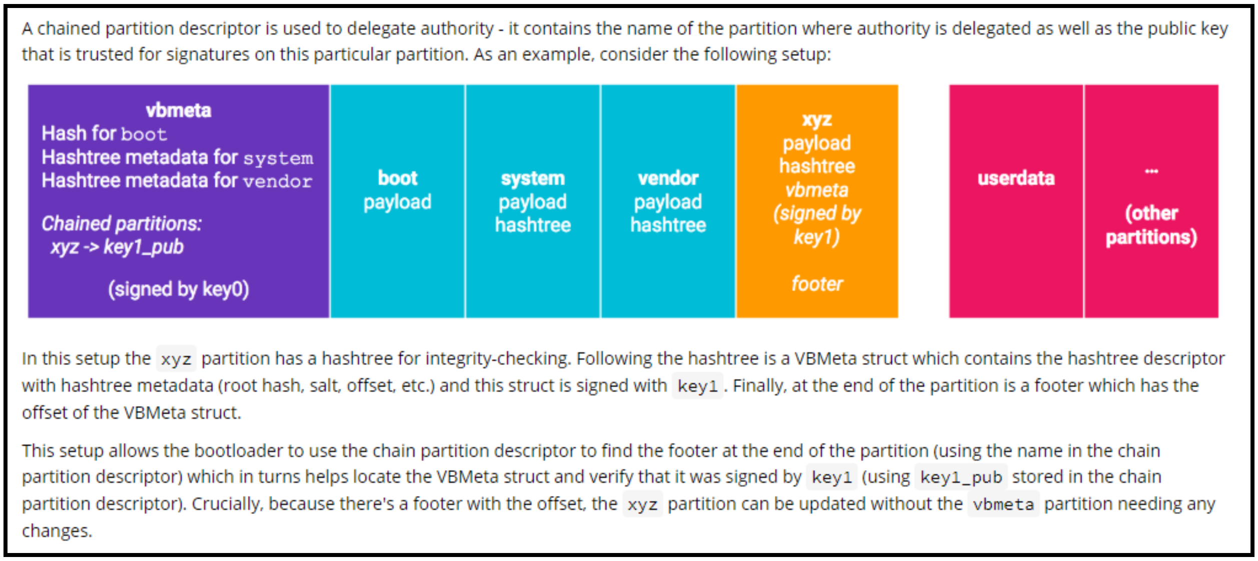
Verified boot is the process of assuring the end user of the integrity of the software running on a device. It typically starts with a read-only portion of the device firmware which loads code and executes it only after cryptographically verifying that the code is authentic and doesn't have any known security flaws. AVB is one implementation of verified boot.

The VBMeta struct

The central data structure used in AVB is the VBMeta struct. This data structure contains a number of descriptors (and other metadata) and all of this data is cryptographically signed. Descriptors are used for image hashes, image hashtree metadata, and so-called *chained partitions*. A simple example is the following:



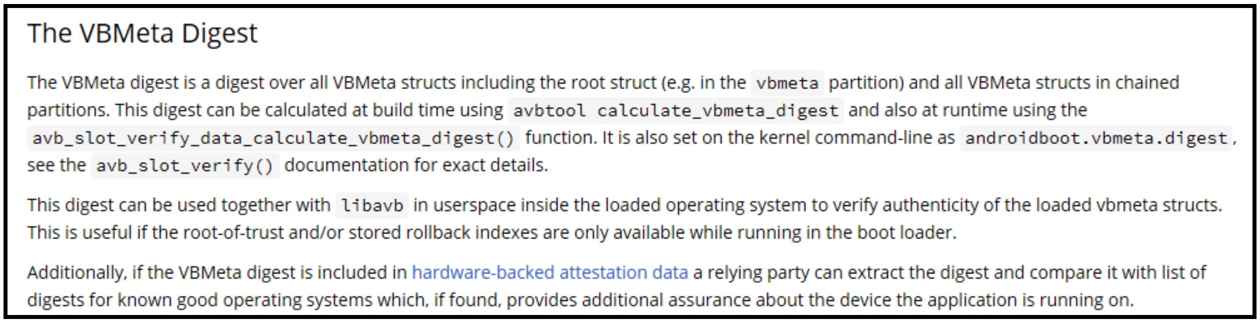
where the `vbmeta` partition holds the hash for the `boot` partition in a hash descriptor. For the `system` and `vendor` partitions a hashtree follows the filesystem data and the `vbmeta` partition holds the root hash, salt, and offset of the hashtree in hashtree descriptors. Because the VBMeta struct in the `vbmeta` partition is cryptographically signed, the boot loader can check the signature and verify it was made by the owner of `key0` (by e.g. embedding the public part of `key0`) and thereby trust the hashes used for `boot` , `system` , and `vendor` .



A chained partition descriptor is used to delegate authority - it contains the name of the partition where authority is delegated as well as the public key that is trusted for signatures on this particular partition. As an example, consider the following setup:

In this setup the `xyz` partition has a hashtree for integrity-checking. Following the hashtree is a VBMeta struct which contains the hashtree descriptor with hashtree metadata (root hash, salt, offset, etc.) and this struct is signed with `key1` . Finally, at the end of the partition is a footer which has the offset of the VBMeta struct.

This setup allows the bootloader to use the chain partition descriptor to find the footer at the end of the partition (using the name in the chain partition descriptor) which in turns helps locate the VBMeta struct and verify that it was signed by `key1` (using `key1_pub` stored in the chain partition descriptor). Crucially, because there's a footer with the offset, the `xyz` partition can be updated without the `vbmeta` partition needing any changes.



The VBMeta Digest

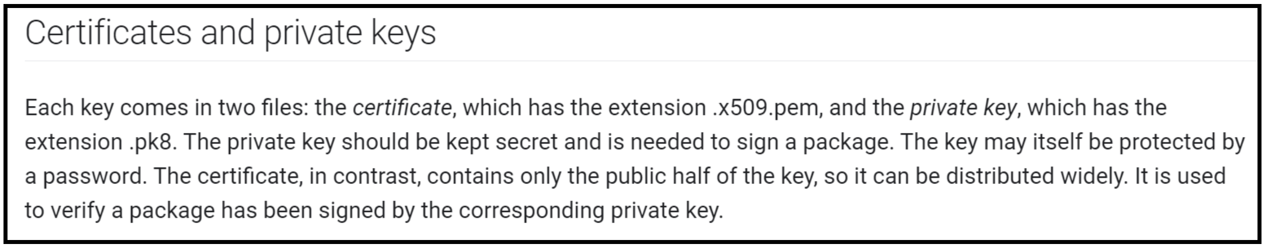
The VBMeta digest is a digest over all VBMeta structs including the root struct (e.g. in the `vbmeta` partition) and all VBMeta structs in chained partitions. This digest can be calculated at build time using `avbtool calculate_vbmeta_digest` and also at runtime using the `avb_slot_verify_data_calculate_vbmeta_digest()` function. It is also set on the kernel command-line as `androidboot.vbmeta.digest` , see the `avb_slot_verify()` documentation for exact details.

This digest can be used together with `libavb` in userspace inside the loaded operating system to verify authenticity of the loaded vbmeta structs. This is useful if the root-of-trust and/or stored rollback indexes are only available while running in the boot loader.

Additionally, if the VBMeta digest is included in hardware-backed attestation data a relying party can extract the digest and compare it with list of digests for known good operating systems which, if found, provides additional assurance about the device the application is running on.

https://android.googlesource.com/platform/external/avb/+/master/README.md#the-vbmeta-digest.

37. Other examples include x509 and/or root certificate authority:

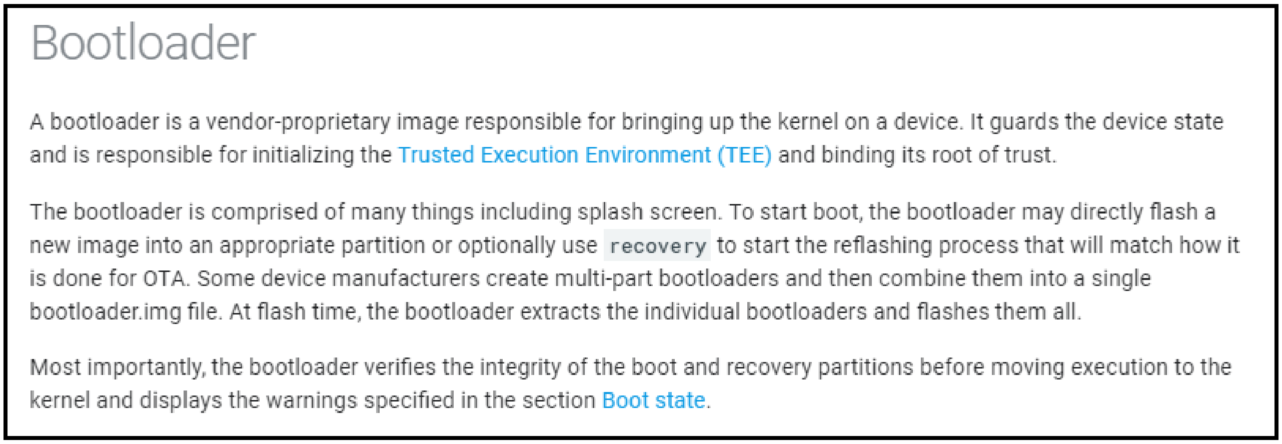


Certificates and private keys

Each key comes in two files: the *certificate*, which has the extension .x509.pem, and the *private key*, which has the extension .pk8. The private key should be kept secret and is needed to sign a package. The key may itself be protected by a password. The certificate, in contrast, contains only the public half of the key, so it can be distributed widely. It is used to verify a package has been signed by the corresponding private key.

https://source.android.com/devices/tech/ota/sign_builds.

38. Once the verification structure has been set up in the BIOS, the VIZIO XR6M10 SmartCast Tablet is configured by VIZIO to reboot, load the OTA update into its volatile memory (e.g., RAM), and then use the at least one license record from the BIOS to verify the OTA update as part of its secure or verified boot process:



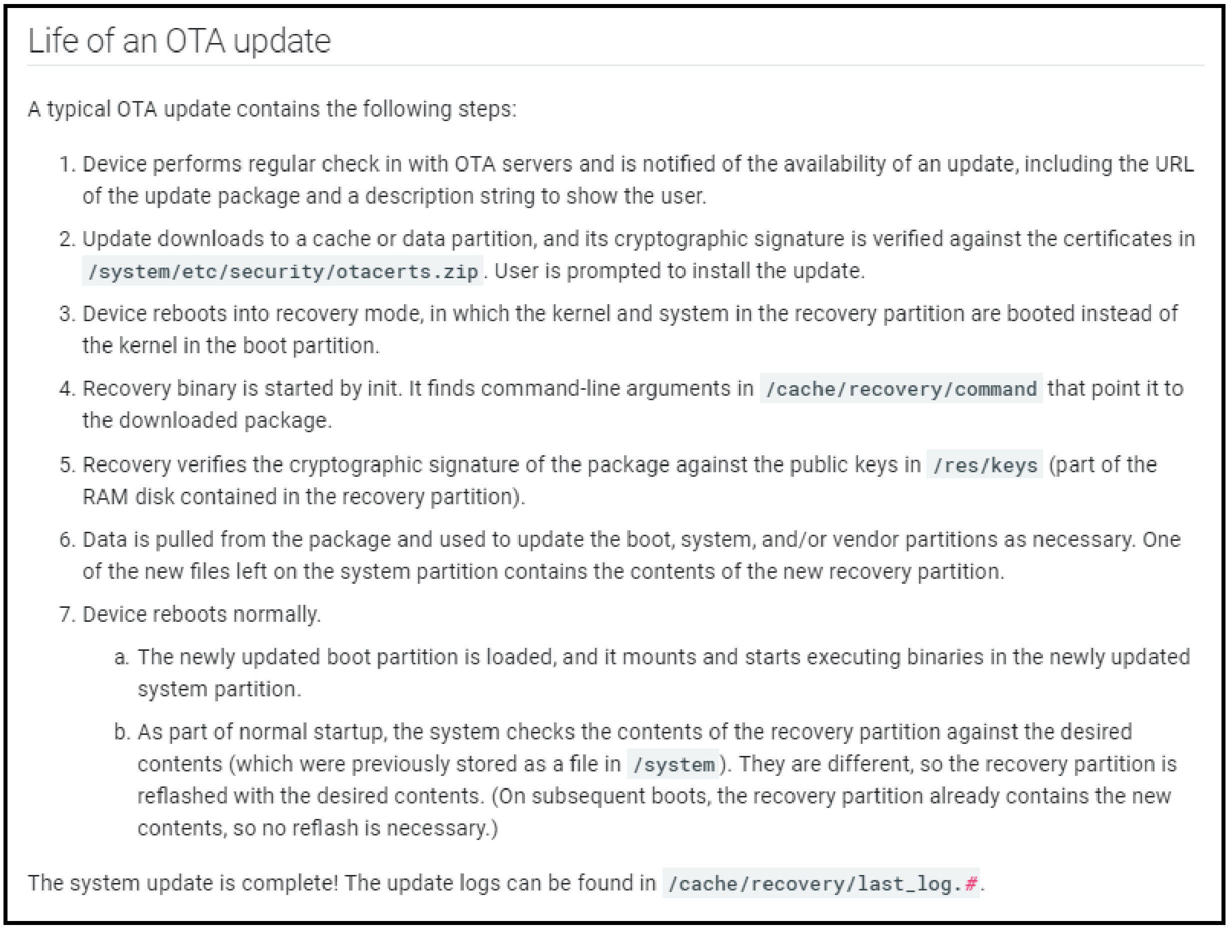
Bootloader

A bootloader is a vendor-proprietary image responsible for bringing up the kernel on a device. It guards the device state and is responsible for initializing the Trusted Execution Environment (TEE) and binding its root of trust.

The bootloader is comprised of many things including splash screen. To start boot, the bootloader may directly flash a new image into an appropriate partition or optionally use `recovery` to start the reflashing process that will match how it is done for OTA. Some device manufacturers create multi-part bootloaders and then combine them into a single bootloader.img file. At flash time, the bootloader extracts the individual bootloaders and flashes them all.

Most importantly, the bootloader verifies the integrity of the boot and recovery partitions before moving execution to the kernel and displays the warnings specified in the section Boot state.

https://source.android.com/devices/bootloader.



Life of an OTA update

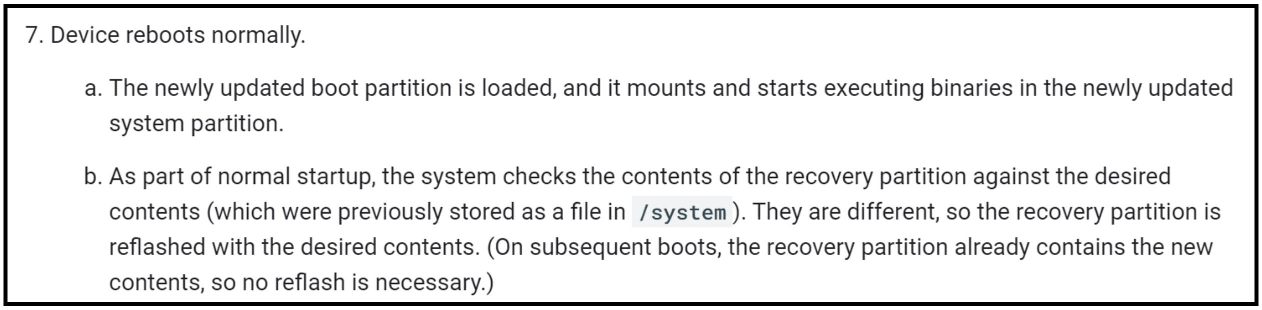
A typical OTA update contains the following steps:

1. Device performs regular check in with OTA servers and is notified of the availability of an update, including the URL of the update package and a description string to show the user.
2. Update downloads to a cache or data partition, and its cryptographic signature is verified against the certificates in `/system/etc/security/otacerts.zip`. User is prompted to install the update.
3. Device reboots into recovery mode, in which the kernel and system in the recovery partition are booted instead of the kernel in the boot partition.
4. Recovery binary is started by init. It finds command-line arguments in `/cache/recovery/command` that point it to the downloaded package.
5. Recovery verifies the cryptographic signature of the package against the public keys in `/res/keys` (part of the RAM disk contained in the recovery partition).
6. Data is pulled from the package and used to update the boot, system, and/or vendor partitions as necessary. One of the new files left on the system partition contains the contents of the new recovery partition.
7. Device reboots normally.
    a. The newly updated boot partition is loaded, and it mounts and starts executing binaries in the newly updated system partition.
    b. As part of normal startup, the system checks the contents of the recovery partition against the desired contents (which were previously stored as a file in `/system`). They are different, so the recovery partition is reflashed with the desired contents. (On subsequent boots, the recovery partition already contains the new contents, so no reflash is necessary.)

The system update is complete! The update logs can be found in `/cache/recovery/last_log.#`.

https://source.android.com/devices/tech/ota/nonab.

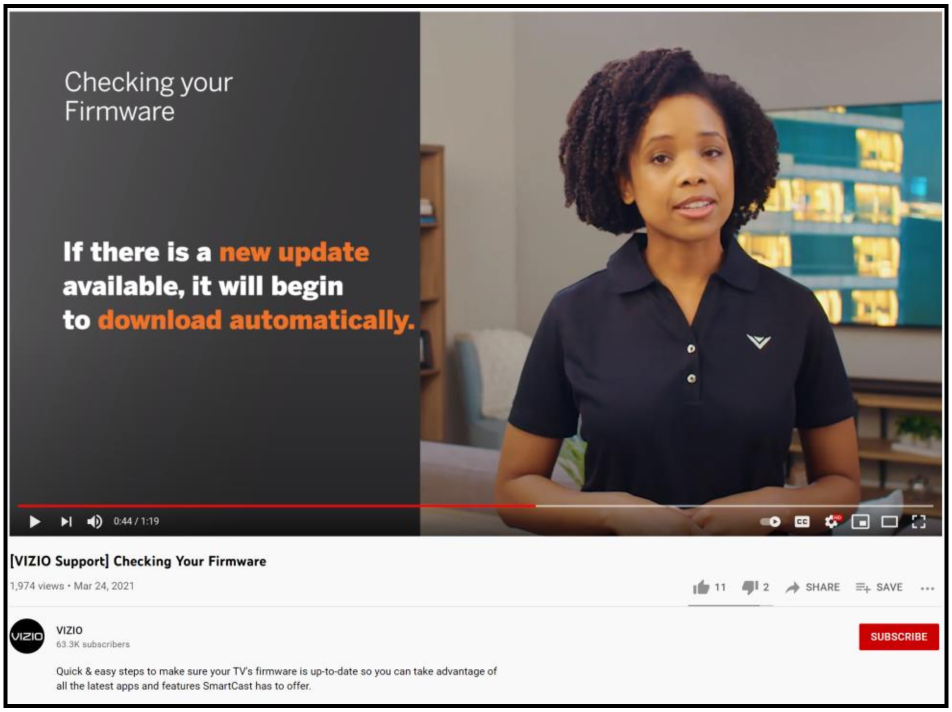
39. If the OTA update is verified, the VIZIO XR6M10 SmartCast Tablet is further configured to load and execute the update.



7. Device reboots normally.
    a. The newly updated boot partition is loaded, and it mounts and starts executing binaries in the newly updated system partition.
    b. As part of normal startup, the system checks the contents of the recovery partition against the desired contents (which were previously stored as a file in `/system`). They are different, so the recovery partition is reflashed with the desired contents. (On subsequent boots, the recovery partition already contains the new contents, so no reflash is necessary.)

https://source.android.com/devices/tech/ota/nonab.

40. Further, during the infringing time period, VIZIO performed or caused to be performed each of the Claim 1 steps identified above by providing an OTA update to each Accused Product, including to various TV products:


Checking your Firmware
If there is a new update available, it will begin to download automatically.
0:44 / 1:19
11
2
SHARE
SAVE
VIZIO
SUBSCRIBE

[VIZIO Support] Checking Your Firmware

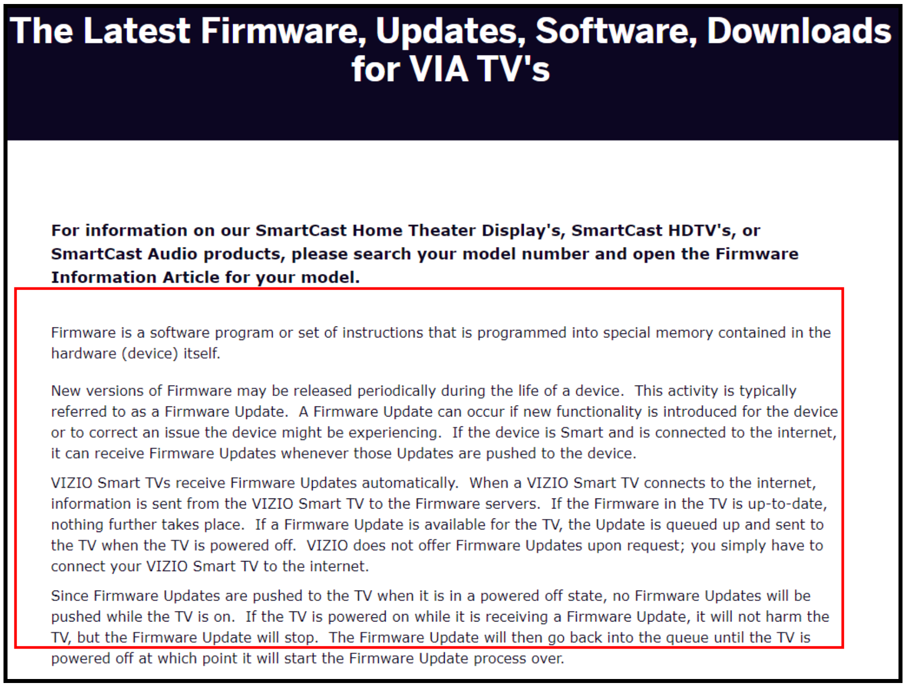
1,974 views • Mar 24, 2021

VIZIO

63.3K subscribers

Quick & easy steps to make sure your TV's firmware is up-to-date so you can take advantage of all the latest apps and features SmartCast has to offer.

https://www.youtube.com/watch?v=AEmzDACKM04



The Latest Firmware, Updates, Software, Downloads for VIA TV's

For information on our SmartCast Home Theater Display's, SmartCast HDTV's, or SmartCast Audio products, please search your model number and open the Firmware Information Article for your model.

Firmware is a software program or set of instructions that is programmed into special memory contained in the hardware (device) itself.

New versions of Firmware may be released periodically during the life of a device. This activity is typically referred to as a Firmware Update. A Firmware Update can occur if new functionality is introduced for the device or to correct an issue the device might be experiencing. If the device is Smart and is connected to the internet, it can receive Firmware Updates whenever those Updates are pushed to the device.

VIZIO Smart TVs receive Firmware Updates automatically. When a VIZIO Smart TV connects to the internet, information is sent from the VIZIO Smart TV to the Firmware servers. If the Firmware in the TV is up-to-date, nothing further takes place. If a Firmware Update is available for the TV, the Update is queued up and sent to the TV when the TV is powered off. VIZIO does not offer Firmware Updates upon request; you simply have to connect your VIZIO Smart TV to the internet.

Since Firmware Updates are pushed to the TV when it is in a powered off state, no Firmware Updates will be pushed while the TV is on. If the TV is powered on while it is receiving a Firmware Update, it will not harm the TV, but the Firmware Update will stop. The Firmware Update will then go back into the queue until the TV is powered off at which point it will start the Firmware Update process over.

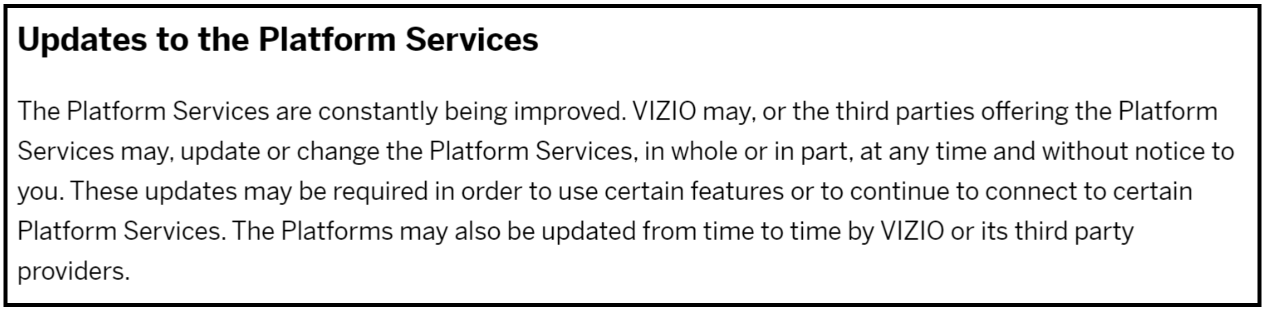
https://support.vizio.com/s/article/The-Latest-Firmware-Updates-Software-Downloads?language=en_US.

41. Further, VIZIO expressly conditions participation in the OTA update process and the receipt of the benefit of a software update on the performance of each of the above steps.

42. Primarily, as described above, VIZIO pre-configures/programs each Accused Product to perform the above described steps upon receiving an OTA update from VIZIO.

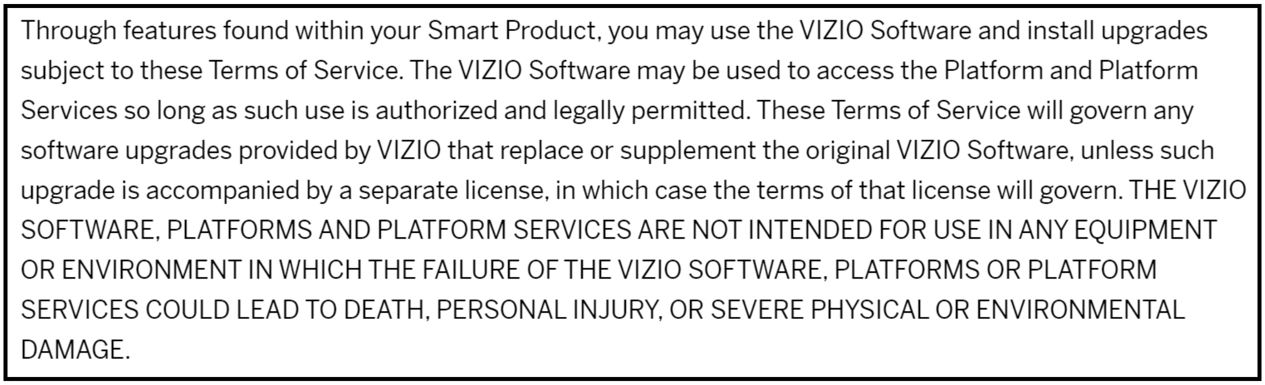
43. Further, VIZIO not only set the time and conditions under which a user could receive and install an OTA update, but VIZIO required all users to accept and install such updates.

44. For example, VIZIO stated the following in its Product Terms of Service:



Updates to the Platform Services

The Platform Services are constantly being improved. VIZIO may, or the third parties offering the Platform Services may, update or change the Platform Services, in whole or in part, at any time and without notice to you. These updates may be required in order to use certain features or to continue to connect to certain Platform Services. The Platforms may also be updated from time to time by VIZIO or its third party providers.



Through features found within your Smart Product, you may use the VIZIO Software and install upgrades subject to these Terms of Service. The VIZIO Software may be used to access the Platform and Platform Services so long as such use is authorized and legally permitted. These Terms of Service will govern any software upgrades provided by VIZIO that replace or supplement the original VIZIO Software, unless such upgrade is accompanied by a separate license, in which case the terms of that license will govern. THE VIZIO SOFTWARE, PLATFORMS AND PLATFORM SERVICES ARE NOT INTENDED FOR USE IN ANY EQUIPMENT OR ENVIRONMENT IN WHICH THE FAILURE OF THE VIZIO SOFTWARE, PLATFORMS OR PLATFORM SERVICES COULD LEAD TO DEATH, PERSONAL INJURY, OR SEVERE PHYSICAL OR ENVIRONMENTAL DAMAGE.

https://www.vizio.com/en/terms/terms-of-service.

45. Further, VIZIO emphasizes the benefits associated with updating the software of its Accused Products, including because such updates are "important to keeping [the device] running smoothly" and to allow users to "take advantage of all the latest apps and features": https://www.youtube.com/watch?v=AEmzDACKM04. VIZIO has also stated that its firmware

updates provide “stability enhancements or . . . correct issues in the event we encounter software related problems”: https://twitter.com/VIZIO/status/1299781696263532547.

46. Further, VIZIO controlled the manner in which each OTA update could be performed, including by pre-configuring each Accused Product such that, upon receiving an OTA update from VIZIO, the device would automatically perform each remaining step of the claimed method.

47. VIZIO also controlled the timing of the performance of such method by determining when to utilize its OTA servers/software to set up a verification structure in each Accused Product.

48. VIZIO also had the right and ability to stop or limit infringement simply by not performing the initial step of using its OTA servers/software to set up a verification structure in each Accused Product. Absent this action by VIZIO, the infringement at issue in this lawsuit would not have occurred.

49. VIZIO’s infringement has caused damage to Ancora, and Ancora is entitled to recover from VIZIO those damages that Ancora has sustained as a result of VIZIO’s infringement.

DEMAND FOR JURY TRIAL

50. Ancora hereby demands a jury trial for all issues so triable.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff prays for judgment as follows:

A. Declaring that VIZIO, Inc. has infringed United States Patent No. 6,411,941 in violation of 35 U.S.C. § 271;

B. Awarding damages to Ancora arising out of this infringement, including enhanced damages pursuant to 35 U.S.C. § 284 and prejudgment and post-judgment interest, in an amount according to proof;

C. Awarding such other costs and relief the Court deems just and proper, including any relief that the Court may deem appropriate under 35 U.S.C. § 285.

Date: July 16, 2021

/s/ Andres Healy
Andres Healy (WA 45578)
SUSMAN GODFREY LLP
1201 Third Avenue, Suite 3800
Seattle, Washington 98101
Tel: (206) 516-3880
Fax: 206-516-3883
ahealy@susmangodfrey.com

Lexie G. White (TX 24048876)
SUSMAN GODFREY LLP
1000 Louisiana Street, Suite 5100
Houston, Texas 77002
Tel: (713) 651-9366
Fax: (713) 654-6666
lwhite@susmangodfrey.com

Charles Ainsworth
State Bar No. 00783521
Robert Christopher Bunt
State Bar No. 00787165
PARKER, BUNT & AINSWORTH, P.C.
100 E. Ferguson, Suite 418
Tyler, TX 75702
903/531-3535
E-mail: charley@pbatyler.com
E-mail: rcbunt@pbatyler.com

COUNSEL FOR PLAINTIFF ANCORA TECHNOLOGIES, INC.